

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-232115

(43)Date of publication of application : 27.08.1999

(51)Int.Cl. G06F 9/445  
G06F 13/00  
G06F 15/16

(21)Application number : 10-296431

(71)Applicant : INTERNATL BUSINESS MACH CORP &lt;IBM&gt;

(22)Date of filing : 19.10.1998

(72)Inventor : MASSEY ANDREW LIAM  
SYED SOHAIL

(30)Priority

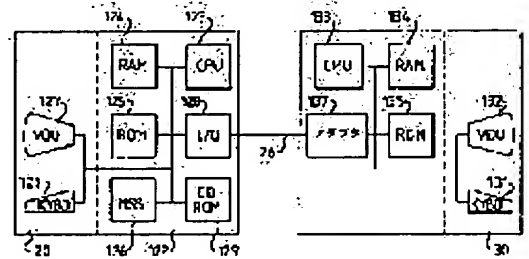
Priority number : 97 9724364 Priority date : 19.11.1997 Priority country : GB

(54) SOFTWARE MODULE TRANSFER METHOD AND DATA PROCESSING NETWORK AND SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To consider the activity of a user in a work station and to reduce the loss of data transmission time between a client system and a server by predicting the type of a software module based on a using history and automatically performing transfer based on the monitoring of a traffic level.

SOLUTION: A server system 20 transmits a first software module type to a connected client system 30. A control software predicts the type of the software module assumed to be requested by the same client based on a data gathering software or compiled history information. The control software decides whether or not LAN traffic is smaller than a certain threshold value, and when it is so, makes a push software push the predicted module to the client 30.



BEST AVAILABLE COPY

## LEGAL STATUS

[Date of request for examination] 27.07.1999

[Date of sending the examiner's decision of rejection] 30.05.2000

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection] 2000-010577

[Date of requesting appeal against examiner's decision of rejection] 12.07.2000

[Date of extinction of right]

\* NOTICES \*

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.\*\*\*\* shows the word which can not be translated.

3. In the drawings, any words are not translated.

## CLAIMS

### [Claim(s)]

[Claim 1] It is the approach of transmitting to preemptive one from host data processing system, in order to perform a software module on a Target data processing system all over a data-processing network. There is a software module of two or more types which offer service of said T-system different by the way, respectively, and in the place which is said host system. The step which determines the type of the 1st software module transmitted to a T-system, The step which predicts the type of the 2nd software module for which said T-system is probably needed by the way next based on the decision of the type of said 1st module, The approach containing the step which transmits said 2nd type of module to said T-system automatically.

[Claim 2] The approach according to claim 1 based on [ including other steps of said server which, by the way, compile the hysteresis of the software-module use in each T-system ] said compiled hysteresis of each T-system in said prediction step.

[Claim 3] The approach containing other steps which transmit said 2nd software module automatically only when smaller than the threshold which supervises the level of the traffic on said network, and has said traffic according to claim 1 or 2.

[Claim 4] An approach given in claim 1 to which said 1st software module answers a demand from said T-system, and is transmitted from said host system thru/or any 1 term of 3.

[Claim 5] An approach given in claim 1 which is the software module which said software module can load dynamically thru/or any 1 term of 4.

[Claim 6] The approach according to claim 5 said software module which can be loaded dynamically is a Java applet for performing through a java usable browser or a java virtual machine on said T-system.

[Claim 7] An approach given in claim 1 said whose network is a local area network thru/or any 1 term of 6.

[Claim 8] An approach given in claim 1 said whose network is the Internet or intranet thru/or any 1 term of 6.

[Claim 9] An approach given in claim 1 said whose 1st software module is said T-system and which offers text-editing service by the way thru/or any 1 term of 8.

[Claim 10] The method according to claim 9 of by the way offering electronic mail communication service that said 2nd software module is said T-system.

[Claim 11] It is a data-processing network containing the server data processing system connected in order to communicate with two or more Target data processing systems. Are in the place of said server data processing system, or it relates to it. A means to transmit in order to perform on it the software module of two or more types which offer service of a T-system different by the way, respectively to said two or more T-systems, A means to supervise discernment of said software module transmitted to said T-system, The decision for the software module of the 1st type to have communicated with the T-system is answered. The software module of the 2nd type predicts that said T-system is probably next needed by the way. A means to make preemptive one transmit said 2nd type of software module to said transfer means to said T-system, and said T-system and by the way A data-processing network including a means to receive said software module from said host system.

[Claim 12] The data-processing [ a data collection means to compile the hysteresis of the software-module use in each aforementioned T-system / further / means / said / prediction / prediction / said ] network according to claim 11 based on said hysteresis of each T-system.

[Claim 13] The data-processing network according to claim 11 or 12 which includes further a means to transmit said 2nd software module automatically only when smaller than the threshold which supervises the level of the traffic on said network, and has said traffic.

[Claim 14] The data-processing network according to claim 11 to 13 which is the software module which the type of each software module can load dynamically.

[Claim 15] The data-processing network according to claim 14 where said software module which can be loaded dynamically is a Java applet, and each T-system contains a java usable browser or a java virtual machine.

[Claim 16] A means to transmit in order to perform on it a means to communicate with two or more client data processing system, and the software module of two or more types which offer service of said T-system different by the way, respectively to said two or more client systems, A means to supervise discernment of said software module transmitted to said T-system, The decision for the software module of the 1st type to have communicated with the T-system is answered. The software module of the 2nd type predicts that said T-system is probably next needed by the way. And server data processing system which includes a means to make said 2nd type of software module transmit to said T-system to preemptive ones in said transfer means.

[Translation done.]

\* NOTICES \*

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.

2. \*\*\*\* shows the word which can not be translated.

3. In the drawings, any words are not translated.

## DETAILED DESCRIPTION

## [Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the approach of transmitting software to preemptive one from a host system all over a data-processing network at a T-system.

[0002]

[Description of the Prior Art] In typical networking environment, two or more client data processing system is connected to one or more server systems. With the 1st general configuration, each client system contains other software by the operating system and option which are stored on the hard file in a client. By the client system, other application software held on the storage relevant to a server system, for example, WORD processing, and database software are accessed, when required.

[0003] In alternative network configuration, by the client system, a client system downloads from a server most or other software which do not have bulk-store capacity at all, but an operating system and a user need in this case, when required, and it memorizes in volatile memory.

[0004] Generally in an existing network, software and other data are transmitted by the demand of a client system from a server. This is automatically performed during activation of the processing specific action by the side of a client user, or on a client. While a server system is using it, the demand to software is waiting for those sequence in the queue, and, thereby, large delay may arise. Furthermore, even when there was no queue, and processed immediately after [ whose demand to software is a server ] by the way receiving, nevertheless in case the demanded software is manufactured and the demanded software is transmitted to a client through a connection link, there is loss of the time amount produced according to the need that a requestor-side client transmits a demand to a server with loss of required time amount. When there is much network traffic, all delay to produce may become large and may have a bad influence on the activity of the user of a client system.

[0005] The technique which predicts the need for the data object of some types by the way it is [ types ] the workstation to which the host system was connected is indicated by the United States patent application No. 5029104. A host answers the data object (for example, electronic mail important for a workstation user) by which a host is received by the way, and includes the heuristics logical circuit which determines possibility that a workstation will use each object. The data object for which a workstation may be used by the way is transmitted to arbitration by the host to a workstation, before a demand is performed by the workstation.

[0006] Therefore, with this conventional technique, a data object is transmitted to preemptive one to a workstation based on the decision that a data object is useful for a workstation user, or for a host system to be important. This decision is due to reception of the data object in a host system, and a user's activity in a workstation is not taken into consideration.

[0007]

[Problem(s) to be Solved by the Invention] According to the 1st mode of this invention, it is the approach of transmitting to preemptive one from host data processing system, in order to perform a software module on a Target date processing system all over a data-processing network. There is a software module of two or more types which offer service of a T-system different by the way, respectively, and in the place which is a host system The step which determines the type of the 1st software module transmitted to a T-system, The step which predicts the type of the 2nd software module of a T-system which is probably needed for the degree by the way based on the decision of the type of the 1st module, The approach containing the step which transmits the module of the 2nd type to a T-system automatically is offered.

[0008] A means to communicate with two or more client data processing system according to the 2nd mode of this invention, A means to transmit in order to perform on it the software module of two or more types which offer service of a T-system different by the way, respectively to two or more client systems, A means to supervise the type of the software module transmitted to the T-system, The decision for the software module of the 1st type to have communicated with the T-system is answered. The software module of the 2nd type predicts that a T-system is probably next needed by the way. And the server data processing system which includes a means to make the software module of the 2nd type transmit to a T-system to preemptive one in a transfer means is offered.

[0009]

[Means for Solving the Problem] Therefore, according to this invention, based on the type of the software module transmitted to the client system, the approach and system which predict about the type of the software module which the client probably needs next are offered. With a desirable configuration, prediction is based on the hysteresis of the software-module use relevant to each client, and this hysteresis is compiled by the server over a fixed time amount period.

[0010] With the desirable operation gestalt of this invention, the level of the traffic on a network is supervised and the module predicted only when smaller than a threshold with the measured traffic is transmitted automatically.

[0011] With one desirable configuration, a software module can be dynamically loaded, in order to perform through a java usable browser or a java virtual machine on a client system (for example, Java applet).

[0012] According to other modes of this invention, it is incorporated into a computer usable medium. In order to perform a software module on a Target date processing system all over a data-processing network It is a computer program product containing the computer usable medium which has a computer read possible program code means to transmit to preemptive one from host data processing system. There is a software module of two or more types which offer service of a T-system different by the way, respectively. A host system by the way, [ a computer read possible program code means ] A computer read possible program code means to determine the type of the 1st software module transmitted to a T-system, A computer read possible program code means to predict the type of the 2nd software module for which the T-system is probably needed by the way next based on the decision of the type of the 1st module, A computer program product including a computer read possible program code means to transmit the module of the 2nd type to a T-system automatically is offered.

[0013] Next, the desirable operation gestalt of this invention is explained only as an example, referring to an attached drawing.

[0014]

[Embodiment of the Invention] If drawing 1 is referred to in the first place first, the local area network (LAN) 10 where the desirable operation gestalt of this invention is carried out is shown in the rough form. The network of drawing 1 can also be instead constituted according to a token ring or other suitable topology, although shown as what is constituted as Ethernet LAN. This invention can be used also in the context of the Internet network or an intranet network again so that it may explain below. In drawing 1, a network contains the server computer system 20 (this operation gestalt IBM PC 730 computer system) connected by the link 26, in order to

communicate with two or more client computer systems 30, 32, 34, and 36. A client computer system is the personal computer which used the microprocessor of the Intel X86 family as the base, or the computer system of other forms. it explains below — as — a client — or it is the "network computer" (for example, IBM network station) containing a Java virtual machine or a Java usable browser. Each client system contains the LAN adapter card or Network Interface Cards (NIC) 38, 40, 42, and 44 which carry out the communication link with a server computer through a link 26.

[0015] Control of the resource on a network including the communication link between a server and a client is carried out using Netware of the Novell shrine which has the "server" component performed on the main processor of a Network Operating System (NOS), for example, server computer system, and the corresponding "requester" component performed on the main processor of each client computer system. In other suitable Network Operating Systems, it is both OS/2 of IBM. A LAN server and OS/2 There is a WARP server. At this operation gestalt, it is Internetwork. Packet An Exchange (IPX) communications protocol is used in relation to a Netware operating system. With other operation gestalten, TCP/IP is a desirable protocol.

[0016] Drawing 2 is the simplified block diagram showing how to connect the server computer system 20 to the client system 30 through a communication link 26. A client system includes the keyboard 131 and display 132 which operate under control of the control-logic circuit which carried out the form of main CPU133 where the system BIOS which contains the system memory (RAM) 134 and the POST code with a system bus was connected to the nonvolatile memory (ROM) 135 stored in it. Further, although a client system is an Ethernet card with this operation gestalt, it contains the network adapter card 137 which is a token ring adapter card with an alternative implementation gestalt. This adapter card contains the nonvolatile memory to which the code used in case it communicates between a client and a server carried out the form of ROM memorized in it.

[0017] The client system of this operation gestalt should care about not having the large capacity storage which carried out the form of a magnetic-disk drive (hard file) for example. Furthermore, when a client user wants to prevent introducing software or data into a client system, as for a client system, it is advantageous not to have the diskette drive, the CD-ROM drive, or the same drive.

[0018] The server computer system of drawing 2 contains the keyboard 121 connected to the system unit 122 containing the large capacity storage 126 which carried out the form of main CPU123, a system RAM 124, a system ROM 125, and two or more magnetic-disk drives that generally consisted of forms of RAID (independent redundancy array of a disk). A server system contains other storage, such as a display 127 (when a network administrator needs to have a dialog a server system and directly) and a diskette drive (not shown), and CD-ROM drive 129, by option. The communication link through a link 26 is carried out by the I/O logical circuit 128 which takes the form of an adapter card.

[0019] On the disk drive of Server RAID, the software module various type with which it differs to download to different various software containing operating system software and a client system, and for a client system use it is stored. These software modules are the comparatively small codes designed so that service of a client system different by the way might be offered. For example, there are an electronic mail, text editing, a calender, a spreadsheet, and a module type designed so that a client user might be provided with other services. A module is based on a specific demand of a user (human being or process), or is transmitted to a client system in the preemptive form by this invention.

[0020] As for a software module, it is advantageous that it can load dynamically on a client system. In this case, when transmitted to a client, a module is loaded automatically, for example, is displayed on a client user in a window. Further, a module can operate once, and it can also be designed so that it may be discarded after that. Or a module is also storable in a cache, in order that a client may use it later. An example of a suitable software module is a Java applet (Java is the trademark of Sun Microsystems). In this case, a client system is Windows. The Java usable browser (for example, Netscape Navigator currently sold from Netscape Communications) or Java virtual machine (JVM) which operates on suitable operating systems, such as NT, is included.

[0021] Reference of drawing 3 shows the interrelation of the group of a software component 200 (this operation gestalt Netware module) who performs on the server system which offers the function of the desirable operation gestalt of this invention. The LAN analysis software 210 (for example, LANAnalyzer currently sold from the Novell company) supervises the traffic on a network in the conventional form, in order to provide a controller 205 with the information about the level of the traffic on LAN. Much suitable LAN analysis software is sold (for example, LANAnalyzer currently sold from the Novell company).

[0022] The data collection software 220 is designed so that it may perform as a background task which creates the hysteresis of use of various software modules on various client systems. This hysteresis is compiled by the data collection task over several days or several weeks, in order to acquire the time of day when the model of the module which the specific client required, and the module were demanded, and the typical sequence for which a different module type is used.

[0023] For example, Client X always requires a calender software module at the beginning of a business day, /or in order [ in order to inspect the time of day of the meeting where existing was planned, and ] to add meeting information further. This specific user always demands a spreadsheet module, in order to update the front financial information or other information about a business day subsequently. Another client Y requires an electronic mail module first, and requires a calender module after that. Please understand that use hysteresis may differ between clients according to the role of each user of the in-house in which LAN is located.

[0024] This hysteresis information is compiled by data collection software, and by this invention, it is used in order to predict about the future requirements for one of client systems. Therefore, this is supervised by the data collection task when Client X requires a calender module. The control software contains the prediction algorithm 225 which predicts the type of the software module which the client system X may use for a degree based on the information from data collection software. In this case, hysteresis shows that Client X surely requires a spreadsheet module after a calender module. In this case, the control software makes Client X push a spreadsheet module, before the demand to the push task 230 from the user of Client X. A transfer of the module predicted next is decided by decision (based on traffic information from a LAN analysis task) to be smaller than the threshold which LAN traffic was defined in advance or was calculated in advance, with a desirable configuration. According to this configuration, the balance of the load on a network is improvable.

[0025] There is no data collection software which creates the hysteresis of use of the module by various clients with an alternative configuration. Instead, prediction is due to Liszt of the relation during the type of a software module. For example, probably, it will be appropriate to a client to predict that the module required of a degree is an electronic mail communication module, when a text editor is required first. An electronic mail communication module is transmitted to a client preemptive one, before a client subsequently requires the module. Modular relation is stored in a server system in a suitable form, for example, the form of a table.

[0026] Although it is ideal to be transmitted just before the predicted use by the client as for the software module predicted

irrespective of prediction mode, a module is dynamically loaded by the client in this case so that the window which displays service may pop up on the screen of a client. Whenever prediction [ of the usable bandwidth on the network in the time of day of elaboratizing of data collection software, and arbitration, and a client user ] possible, it depends for the capacity to realize this ideal configuration. the module which can be loaded dynamically is discarded after use — having (a module being broadcast again by the server in this case) — or a consecutive call is predicted and it is stored in a cache.

[0027] It cannot be necessary to load a module dynamically and it is rather transmitted to the cache memory of the place of a client for the call of consecutiveness by the user. The suitable configuration of arbitration can be used and the cash advance of the software module in a client can be controlled. For example, when an usable cache is full, a software module is purged by the FIFO method, or each module relates the priority (for example, a text editor is more important than a spreadsheet) with it, and the tooth space in a cache can be released according to a module priority.

[0028] Next, reference of drawing 4 shows the flow chart of the step relevant to the desirable approach of this invention. This approach begins from the place of 300 like illustration. A server system is step 302 and is transmitted to one of the client systems to which the 1st software-module type was connected. This is based on the demand of a client or is carried out by the server in a preemptive form. The control software is step 304 and predicts about the type of the software module which will probably be demanded by the same client based on the hysteresis information compiled by data collection software. At step 306, a decision about whether the control software is smaller than a threshold with LAN traffic is made, and if that is right, the client in question will be made to push the module which is step 308 and was predicted for push software. This process is ended at step 310.

[0029] As already shown, the use hysteresis relevant to a specific client is peculiar to the user of the client. A success of a prediction algorithm is dependent on a client system being used by the same user. This is so also in the opening office environment which generally contains a workstation group. Generally the specific system which should work on it is assigned to the individual in such office. Therefore, when User A uses a workstation WS 10 on Monday every week, it describes that the pattern of data collection software of software-module use is comparatively fixed from Monday to Monday. Therefore, a server system is predicted before User A arrives on Monday morning, and it supplies a suitable software module to a workstation WS 1. Similarly, into the time amount of the remainder of the day, a server is predicted before the demand from a workstation WS 1, and it supplies other software modules.

[0030] When the demand to the software module which chooses that User B uses a workstation WS 10 on Monday morning by a certain reason, and is not usually demanded by WS10 is performed, data collection software describes this. Instead of predicting and supplying the usual "Monday" sequence of a software module based on this, the control software stops predicting or predicts software use based on software-module type relation purely in an above-mentioned form.

[0031] This invention can be used also in the context of the Internet or intranet again. Drawing 5 is the schematic diagram of such a configuration of that the host system 400 connected to two or more client systems in which three of them are shown as systems 412, 414, and 416 through the Internet or intranet 410 is included. A host system has the same configuration as the host of drawing 1. A client system contains the Java virtual machine which performs the Java applet software module received from the Java usable browser 420 or the host system. Each client contains the cache 422 which memorizes a software module temporarily by option. A host system can compile and maintain the hysteresis of module use with reference to each IP address of a client system.

[0032] As mentioned above, although the desirable operation gestalt has been explained about the client system which does not have large capacity storage, he can understand that this invention is applicable also to the client system which has some nonvolatile storage or same storage of software and/or data, such as a hard file memorized locally.

[0033] Furthermore, with a desirable operation gestalt, the module with which the single server system carried out and predicted the monitoring function and the prediction function is transmitted to a client. With an alternative configuration, by the way, one server lengthens a monitor step and a prediction step, and, by the way, as for storage and transfer of a carrier eclipse and a software module, the 2nd server is taken over.

[0034] As a conclusion, the following matters are indicated about the configuration of this invention.

[0035] (1) It is the approach of transmitting to preemptive one from host data processing system, in order to perform a software module on a Target date processing system all over a data-processing network. There is a software module of two or more types which offer service of said T-system different by the way, respectively, and in the place which is said host system The step which determines the type of the 1st software module transmitted to a T-system, The step which predicts the type of the 2nd software module for which said T-system is probably needed by the way next based on the decision of the type of said 1st module, The approach containing the step which transmits said 2nd type of module to said T-system automatically.

(2) An approach given in the above (1) based on [ including other steps of said server which, by the way, compile the hysteresis of the software-module use in each T-system ] said compiled hysteresis of each T-system in said prediction step.

(3) An approach the above (1) containing other steps which transmit said 2nd software module automatically only when smaller than the threshold which supervises the level of the traffic on said network, and has said traffic, or given in (2).

(4) The above (1) to which said 1st software module answers a demand from said T-system, and is transmitted from said host system thru/or an approach given in any 1 term of (3).

(5) An approach the above (1) which is the software module which said software module can load dynamically thru/or given in any 1 term of (4).

(6) An approach given in the above (5) said whose software module which can be loaded dynamically is a Java applet for performing through a java usable browser or a java virtual machine on said T-system.

(7) The above (1) said whose network is a local area network thru/or an approach given in any 1 term of (6).

(8) The above (1) said whose network is the Internet or intranet thru/or an approach given in any 1 term of (6).

(9) The above (1) said whose 1st software module is said T-system and which offers text-editing service by the way thru/or an approach given in any 1 term of (8).

(10) An approach given in the above (9) said whose 2nd software module is said T-system and which, by the way, offers electronic mail communication service.

(11) It is a data-processing network containing the server data processing system connected in order to communicate with two or more Target date processing systems. Are in the place of said server data processing system, or it relates to it. A means to transmit in order to perform on it the software module of two or more types which offer service of a T-system different by the way, respectively to said two or more T-systems, A means to supervise discernment of said software module transmitted to said T-system, The decision

for the software module of the 1st type to have communicated with the T-system is answered. The software module of the 2nd type predicts that said T-system is probably next needed by the way. A means to make preemptive one transmit said 2nd type of software module to said transfer means to said T-system, and said T-system and by the way A data-processing network including a means to receive said software module from said host system.

(12) A data-processing network given in said prediction means' above [ a data collection means to compile the hysteresis of the software-module use in each aforementioned T-system / further ] (11) based on said hysteresis of each T-system of said prediction.

(13) The above (11) which includes further a means to transmit said 2nd software module automatically only when smaller than the threshold which supervises the level of the traffic on said network, and has said traffic, or a data-processing network given in (12).

(14) The above (11) which is the software module which the type of each software module can load dynamically thru/or a data-processing network given in either of (13).

(15) A data-processing network given in the above (14) each T-system of whose said software module which can be loaded dynamically is a Java applet, and contains a java usable browser or a java virtual machine.

(16) A means to communicate with two or more client data processing system, A means to transmit in order to perform on it the software module of two or more types which offer service of said T-system different by the way, respectively to said two or more client systems, A means to supervise discernment of said software module transmitted to said T-system, The decision for the software module of the 1st type to have communicated with the T-system is answered. The software module of the 2nd type predicts that said T-system is probably next needed by the way. And server data processing system which includes a means to make said 2nd type of software module transmit to said T-system to preemptive one in said transfer means.

---

[Translation done.]

\* NOTICES \*

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.\*\*\*\* shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

---

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the schematic diagram of the data-processing network containing the host system connected to two or more client data processing system.

[Drawing 2] It is the schematic diagram of the main components of the client system of drawing 1 , and a host system.

[Drawing 3] It is the schematic diagram showing the dialogue between the elements of the host system used in the desirable operation gestalt of this invention.

[Drawing 4] It is the flow chart showing the step of 1 desirable operation gestalt of this invention.

[Drawing 5] This invention is the schematic diagram of the alternative network configuration used with it.

[Description of Notations]

- 10 Local Area Network (LAN)
- 20 Server Computer System
- 26 Communication Link
- 30 Client Computer System
- 32 Client Computer System
- 34 Client Computer System
- 36 Client Computer System
- 38 Network Interface Card
- 40 Network Interface Card
- 42 Network Interface Card
- 44 Network Interface Card
- 121 Keyboard
- 122 System Unit
- 123 The Main CPU
- 124 System RAM
- 125 System ROM
- 126 Large Capacity Storage
- 127 Display
- 128 I/O Logical Circuit
- 129 CD-ROM Drive
- 131 Keyboard
- 132 Display
- 133 The Main CPU
- 134 System Memory (RAM)
- 135 Nonvolatile Memory (ROM)

- 137 Network Adapter Card
- 200 Software Component
- 205 Controller
- 210 LAN Analysis Software
- 220 Data Collection Software
- 225 Prediction Algorithm
- 230 Push Task
- 400 Host System
- 410 Internet or Intranet
- 412 Client System
- 414 Client System
- 416 Client System
- 420 Java Usable Browser
- 422 Cache

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-232115

(43) 公開日 平成11年(1999) 8月27日

(51) Int.Cl.<sup>6</sup>  
G 0 6 F 9/445  
13/00  
15/16

識別記号  
3 5 1  
6 2 0

F I  
G 0 6 F 9/06  
13/00  
15/16

4 2 0 C  
3 5 1 H  
6 2 0 B

審査請求 未請求 請求項の数16 O L (全 10 頁)

(21) 出願番号 特願平10-296431  
(22) 出願日 平成10年(1998)10月19日  
(31) 優先権主張番号 9 7 2 4 3 6 4 . 6  
(32) 優先日 1997年11月19日  
(33) 優先権主張国 イギリス (G B)

(71) 出願人 390009531  
インターナショナル・ビジネス・マシーンズ・コーポレーション  
INTERNATIONAL BUSIN  
ESS MASCHINES CORPO  
RATION  
アメリカ合衆国10504、ニューヨーク州  
アーモンク (番地なし)  
(72) 発明者 アンドリュウ・リーム・マッセイ  
イギリス ケー・エー30 9ビー・ワイ  
アーシュレー ラーガス パーンサイド・  
ロード 61  
(74) 代理人 弁理士 坂口 博 (外1名)

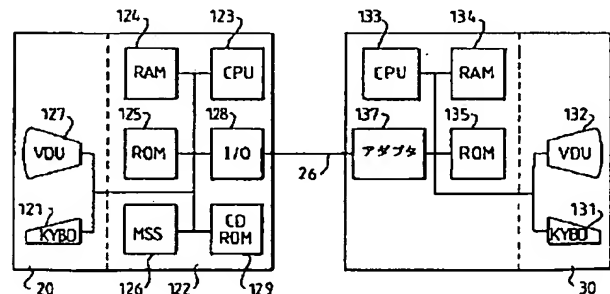
最終頁に続く

(54) 【発明の名称】 ソフトウェア・モジュール転送方法、データ処理ネットワークおよびシステム

(57) 【要約】

【課題】 データ処理ネットワーク中でソフトウェア・モジュール (例えば j a v a アプレットなど動的にロードできるコード) をクライアント・データ処理システム上で実行するためにホスト・データ処理システムからブリエンプティブに転送する方法およびシステムを提供すること。

【解決手段】 ソフトウェア・モジュールは、それぞれクライアント・システムのところで異なるサービス (例えば電子メール、テキスト・エディタなど) を提供する異なる多数のタイプである。ホスト・システムは、クライアントに転送される第1のソフトウェア・モジュールのタイプを決定し、この決定に基づいて、次にクライアントのところで必要とされるであろうソフトウェア・モジュールのタイプに関して予測を行う。次いで、予測したソフトウェア・モジュールが、クライアントから同じものを要求される前にクライアント・システムに転送される。





## 【特許請求の範囲】

【請求項1】データ処理ネットワーク中でソフトウェア・モジュールをターゲット・データ処理システム上で実行するためにホスト・データ処理システムからクライアントに転送する方法であって、それぞれ前記ターゲット・システムのところで異なるサービスを提供する複数のタイプのソフトウェア・モジュールがあり、前記ホスト・システムのところで、ターゲット・システムに転送される第1のソフトウェア・モジュールのタイプを決定するステップと、

前記第1のモジュールのタイプの決定に基づいて、次に前記ターゲット・システムのところで必要とされるであろう第2のソフトウェア・モジュールのタイプを予測するステップと、

前記第2のタイプのモジュールを前記ターゲット・システムに自動的に転送するステップとを含む方法。

【請求項2】前記サーバのところで、各ターゲット・システムにおけるソフトウェア・モジュール使用の履歴をコンパイルする他のステップを含み、前記予測ステップが各ターゲット・システムの前記コンパイルされた履歴に基づく請求項1に記載の方法。

【請求項3】前記ネットワーク上のトラフィックのレベルを監視し、かつ前記トラフィックがあるしきい値よりも小さいときだけ前記第2のソフトウェア・モジュールを自動的に転送する他のステップを含む請求項1または請求項2に記載の方法。

【請求項4】前記第1のソフトウェア・モジュールが前記ターゲット・システムからの要求に回答して、前記ホスト・システムから転送される請求項1ないし3のいずれか一項に記載の方法。

【請求項5】前記ソフトウェア・モジュールが動的にロードできるソフトウェア・モジュールである請求項1ないし4のいずれか一項に記載の方法。

【請求項6】前記動的にロードできるソフトウェア・モジュールが、前記ターゲット・システム上でjava使用可能ブラウザまたはjava仮想マシンを介して実行するためのJavaアプレットである請求項5に記載の方法。

【請求項7】前記ネットワークがローカル・エリア・ネットワークである請求項1ないし6のいずれか一項に記載の方法。

【請求項8】前記ネットワークがインターネットまたはイントラネットである請求項1ないし6のいずれか一項に記載の方法。

【請求項9】前記第1のソフトウェア・モジュールが前記ターゲット・システムのところでテキスト編集サービスを提供する請求項1ないし8のいずれか一項に記載の方法。

【請求項10】前記第2のソフトウェア・モジュールが前記ターゲット・システムのところで電子メール通信サ

ービスを提供する請求項9に記載の方法。

【請求項11】複数のターゲット・データ処理システムと通信するために接続されたサーバ・データ処理システムを含むデータ処理ネットワークであって、

前記サーバ・データ処理システムのところにあるか、またはそれに関連し、それぞれターゲット・システムのところで異なるサービスを提供する複数のタイプのソフトウェア・モジュールを前記複数のターゲット・システムにその上で実行するために転送する手段と、

10 前記ターゲット・システムに送信された前記ソフトウェア・モジュールの識別を監視する手段と、

第1のタイプのソフトウェア・モジュールがターゲット・システムと通信していたという決定に回答して、第2のタイプのソフトウェア・モジュールが次に前記ターゲット・システムのところで必要とされるであろうと予測し、かつ前記転送手段に前記第2のタイプのソフトウェア・モジュールを前記ターゲット・システムにクライアントに転送させる手段と、

20 前記ターゲット・システムのところで、前記ホスト・システムからの前記ソフトウェア・モジュールを受信する手段とを含むデータ処理ネットワーク。

【請求項12】各前記ターゲット・システムにおけるソフトウェア・モジュール使用の履歴をコンパイルするデータ収集手段をさらに含み、前記予測手段が前記予測を各ターゲット・システムの前記履歴を基にする請求項11に記載のデータ処理ネットワーク。

30 【請求項13】前記ネットワーク上のトラフィックのレベルを監視し、かつ前記トラフィックがあるしきい値よりも小さいときだけ前記第2のソフトウェア・モジュールを自動的に転送する手段をさらに含む請求項11または請求項12に記載のデータ処理ネットワーク。

【請求項14】各ソフトウェア・モジュールのタイプが動的にロードできるソフトウェア・モジュールである請求項11ないし請求項13のいずれかに記載のデータ処理ネットワーク。

40 【請求項15】前記動的にロードできるソフトウェア・モジュールがJavaアプレットであり、各ターゲット・システムがjava使用可能ブラウザまたはjava仮想マシンを含む請求項14に記載のデータ処理ネットワーク。

【請求項16】複数のクライアント・データ処理システムと通信する手段と、

それぞれ前記ターゲット・システムのところで異なるサービスを提供する複数のタイプのソフトウェア・モジュールを前記複数のクライアント・システムにその上で実行するために転送する手段と、

前記ターゲット・システムに送信された前記ソフトウェア・モジュールの識別を監視する手段と、

50 第1のタイプのソフトウェア・モジュールがターゲット・システムと通信していたという決定に回答して、第2

のタイプのソフトウェア・モジュールが次に前記ターゲット・システムのところで必要とされるであろうと予測し、かつ前記転送手段に前記第2のタイプのソフトウェア・モジュールを前記ターゲット・システムにブリエンプティブに転送させる手段とを含むサーバ・データ処理システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、データ処理ネットワーク中でソフトウェアをホスト・システムからターゲット・システムにブリエンプティブに転送する方法に関する。

【0002】

【従来の技術】代表的なネットワーク環境中では、複数のクライアント・データ処理システムが1つまたは複数のサーバ・システムに接続される。第1の一般的な構成では、各クライアント・システムは、クライアント中のハード・ファイル上に格納されるオペレーティング・システムおよび任意選択で他のソフトウェアを含む。サーバ・システムに関連する記憶装置上に保持される他のアプリケーション・ソフトウェア、例えばワード・プロセッシング、データベース・ソフトウェアは、クライアント・システムによって必要なときにアクセスされる。

【0003】代替ネットワーク構成では、クライアント・システムは、ほとんどまたはまったく大容量記憶能力を有せず、この場合、オペレーティング・システムおよびユーザが必要とする他のソフトウェアは、クライアント・システムによって必要なときにサーバからダウンロードされ、揮発性メモリ中に記憶される。

【0004】現存のネットワークでは、ソフトウェアおよび他のデータは、一般に、クライアント・システムの要求によりサーバから転送される。これは、クライアント・ユーザ側の特定のアクションか、またはクライアント上での処理の実行中に自動的に行われる。サーバ・システムが使用中のとき、ソフトウェアに対する要求は、待ち行列中でそれらの順番を待っており、それにより大きい遅延が生じることがある。さらに、待ち行列がまったくなく、かつソフトウェアに対する要求がサーバのところで受信された直後に処理された場合でも、それにもかかわらず、要求されたソフトウェアを製造し、要求されたソフトウェアを接続リンクを介してクライアントに伝達する際に必要な時間の損失と共に要求側クライアントが要求をサーバに送信する必要によって生じる時間の損失がある。ネットワーク・トラフィックが多いとき、生じる全遅延は大きくなり、クライアント・システムのユーザの活動に悪影響を及ぼすことがある。

【0005】米国特許出願第5029104号には、ホスト・システムが接続されたワークステーションのところでいくつかのタイプのデータ・オブジェクトの必要を

予測する技法が記載されている。ホストは、ホストのところで受信されるデータ・オブジェクト（例えば、ワークステーション・ユーザにとって重要な電子メール）に応答して、ワークステーションが各オブジェクトを使用する可能性を決定するヒューリスティック論理回路を含む。ワークステーションのところで使用される可能性のあるデータ・オブジェクトは、ワークステーションによって要求が行われる前にホストによって任意にワークステーションに送信される。

【0006】したがってこの従来技術では、ホスト・システムは、データ・オブジェクトがワークステーション・ユーザにとって有用または重要であるという決定に基づいてデータ・オブジェクトをワークステーションにブリエンプティブに送信する。この決定は、ホスト・システムにおけるデータ・オブジェクトの受信に基づいており、ワークステーションにおけるユーザの活動を考慮しない。

【0007】

【発明が解決しようとする課題】本発明の第1の態様によれば、データ処理ネットワーク中でソフトウェア・モジュールをターゲット・データ処理システム上で実行するためにホスト・データ処理システムからブリエンプティブに転送する方法であって、それぞれターゲット・システムのところで異なるサービスを提供する複数のタイプのソフトウェア・モジュールがあり、ホスト・システムのところで、ターゲット・システムに転送される第1のソフトウェア・モジュールのタイプを決定するステップと、第1のモジュールのタイプの決定に基づいて、ターゲット・システムのところで次に必要とされるであろう第2のソフトウェア・モジュールのタイプを予測するステップと、第2のタイプのモジュールをターゲット・システムに自動的に転送するステップとを含む方法が提供される。

【0008】本発明の第2の態様によれば、複数のクライアント・データ処理システムと通信する手段と、それぞれターゲット・システムのところで異なるサービスを提供する複数のタイプのソフトウェア・モジュールを複数のクライアント・システムにその上で実行するために転送する手段と、ターゲット・システムに送信されたソフトウェア・モジュールのタイプを監視する手段と、第1のタイプのソフトウェア・モジュールがターゲット・システムと通信していたという決定に応答して、第2のタイプのソフトウェア・モジュールが次にターゲット・システムのところで必要とされるであろうと予測し、かつ転送手段に第2のタイプのソフトウェア・モジュールをターゲット・システムにブリエンプティブに転送させる手段とを含むサーバ・データ処理システムが提供される。

【0009】

【課題を解決するための手段】したがって本発明によれ

ば、クライアント・システムに転送されたソフトウェア・モジュールのタイプに基づいて、次にクライアントが必要とするであろうソフトウェア・モジュールのタイプに関して予測を行う方法およびシステムが提供される。好ましい構成では、予測は、各クライアントに関連するソフトウェア・モジュール使用の履歴に基づいており、この履歴は、一定の時間期間にわたってサーバによってコンパイルされる。

【0010】本発明の好ましい実施形態では、ネットワーク上のトラフィックのレベルを監視し、測定したトラフィックがあるしきい値よりも小さいときだけ予測したモジュールを自動的に転送する。

【0011】1つの好ましい構成では、ソフトウェア・モジュールは、クライアント・システム上でjava使用可能ブラウザまたはjava仮想マシンを介して実行するために動的にロードできる（例えば、Javaアプレット）。

【0012】本発明の他の態様によれば、コンピュータ使用可能媒体中に組み込まれ、データ処理ネットワーク中でソフトウェア・モジュールをターゲット・データ処理システム上で実行するためにホスト・データ処理システムからブリエンプティブに転送するコンピュータ読取り可能プログラム・コード手段を有するコンピュータ使用可能媒体を含むコンピュータ・プログラム製品であって、それぞれターゲット・システムのところで異なるサービスを提供する複数のタイプのソフトウェア・モジュールがあり、コンピュータ読取り可能プログラム・コード手段が、ホスト・システムのところで、ターゲット・システムに転送される第1のソフトウェア・モジュールのタイプを決定するコンピュータ読取り可能プログラム・コード手段と、第1のモジュールのタイプの決定に基づいて、次にターゲット・システムのところで必要とされるであろう第2のソフトウェア・モジュールのタイプを予測するコンピュータ読取り可能プログラム・コード手段と、第2のタイプのモジュールをターゲット・システムに自動的に転送するコンピュータ読取り可能プログラム・コード手段とを含むコンピュータ・プログラム製品が提供される。

【0013】次に、本発明の好ましい実施形態について、添付の図面を参照しながら例としてのみ説明する。

【0014】

【発明の実施の形態】まず第一に図1を参照すると、本発明の好ましい実施形態が実施されるローカル・エリア・ネットワーク（LAN）10が概略的な形で示されている。図1のネットワークは、イーサネットLANとして構成されるものとして示されているが、代わりにトークン・リングまたは他の適切なトポロジに従って構成することもできる。以下で説明するように、本発明はまた、インターネット・ネットワークまたはイントラネット・ネットワークのコンテキストにおいても使用でき

る。図1において、ネットワークは、複数のクライアント・コンピュータ・システム30、32、34、36と通信するためにリンク26によって接続されたサーバ・コンピュータ・システム20（この実施形態では、IBM PC 730コンピュータ・システム）を含む。クライアント・コンピュータ・システムは、インテルX86ファミリーのマイクロプロセッサをベースにしたパーソナル・コンピュータまたは他の形のコンピュータ・システムである。以下で説明するように、クライアントはあるいはJava仮想マシンまたはJava使用可能ブラウザを含む「ネットワーク・コンピュータ」（例えばIBMネットワーク・ステーション）である。各クライアント・システムは、リンク26を介してサーバ・コンピュータとの通信を実施するLANアダプタ・カードまたはネットワーク・インタフェース・カード（NIC）38、40、42、44を含む。

【0015】サーバとクライアントとの間の通信を含むネットワーク上の資源の制御は、ネットワーク・オペレーティング・システム（NOS）、例えば、サーバ・コンピュータ・システムの主プロセッサ上で実行する「サーバ」構成要素と、各クライアント・コンピュータ・システムの主プロセッサ上で実行する対応する「リクエスト」構成要素とを有するNovell社のNetwareを使用して実施される。他の適切なネットワーク・オペレーティング・システムには、どちらもIBM社のOS/2 LANサーバおよびOS/2 WARPサーバがある。この実施形態では、Internetwork Packet Exchange（IPX）通信プロトコルがNetwareオペレーティング・システムに関連して使用される。他の実施形態では、TCP/IPが好ましいプロトコルである。

【0016】図2は、通信リンク26を介してサーバ・コンピュータ・システム20をクライアント・システム30に接続する方法を示す簡略化されたブロック図である。クライアント・システムは、システム・バスによってシステム・メモリ（RAM）134およびPOSTコードを含むシステムBIOSがその中に格納された不揮発性メモリ（ROM）135に接続された主CPU133の形をした制御論理回路の制御下で動作するキーボード131およびディスプレイ132を含む。クライアント・システムはさらに、この実施形態ではイーサネット・カードであるが、代替実施形態ではトークン・リング・アダプタ・カードであるネットワーク・アダプタ・カード137を含む。このアダプタ・カードは、クライアントとサーバとの間で通信を実施する際に使用されるコードがその中に記憶されたROMの形をした不揮発性メモリを含む。

【0017】この実施形態のクライアント・システムは、例えば、磁気ディスク・ドライブ（ハード・ファイル）の形をした大容量記憶装置を備えていないことに留

意されたい。さらに、クライアント・ユーザがソフトウェアまたはデータをクライアント・システム中に導入するのを防ぎたい場合、クライアント・システムは、ディスク・ドライブ、CD-ROMドライブ、または同様のドライブを備えていないことが有利である。

【0018】図2のサーバ・コンピュータ・システムは、主CPU123、システムRAM124、システムROM125、および一般にRAID（独立したディスクの冗長アレイ）の形で構成された複数の磁気ディスク・ドライブの形をした大容量記憶装置126を含むシステム装置122に接続されたキーボード121を含む。サーバ・システムは、任意選択で、ディスプレイ127（ネットワーク管理者がサーバ・システムと直接対話する必要がある場合）およびディスク・ドライブ（図示せず）やCD-ROMドライブ129など他の記憶装置を含む。リンク26を介する通信は、アダプタ・カードの形をとる入出力論理回路128によって実施される。

【0019】サーバRAIDのディスク・ドライブ上には、オペレーティング・システム・ソフトウェアを含む異なる様々なソフトウェア、およびクライアント・システムにダウンロードし、またクライアント・システムが使用するための異なる様々なタイプのソフトウェア・モジュールが格納される。これらのソフトウェア・モジュールは、クライアント・システムのところで異なるサービスを提供するように設計された比較的小さいコードである。例えば、電子メール、テキスト編集、カレンダー、スプレッドシート、および他のサービスをクライアント・ユーザに提供するように設計されたモジュール・タイプがある。モジュールは、ユーザ（人間またはプロセス）の特定の要求によるか、または本発明によるブリエンプティブな形でクライアント・システムに転送される。

【0020】ソフトウェア・モジュールは、クライアント・システム上に動的にロードできることが有利である。この場合、モジュールは、クライアントに転送されたとき、自動的にロードし、例えばウィンドウ内でクライアント・ユーザに表示される。モジュールはさらに、一度動作し、その後廃棄されるように設計することもできる。あるいは、モジュールは、後でクライアントが使用するためにキャッシュに格納することもできる。適切なソフトウェア・モジュールの一例は、Javaアプレットである（JavaはSun Microsystems社の登録商標である）。この場合、クライアント・システムは、Windows NTなど適切なオペレーティング・システム上で動作するJava使用可能ブラウザ（例えばNetscape Communications社から販売されているNetscape Navigator）またはJava仮想マシン（JVM）を含む。

【0021】図3を参照すると、本発明の好ましい実施形態の機能を提供するサーバ・システム上で実行するソフトウェア・コンポーネント200（この実施形態ではNetworkwareモジュール）のグループの相互関係が示されている。LAN解析ソフトウェア210（例えば、Novell社から販売されているLANalyzer）は、LAN上のトラフィックのレベルに関する情報をコントローラ205に提供するために従来の形でネットワーク上のトラフィックを監視する。多数の適切なLAN解析ソフトウェアが販売されている（例えば、Novell社から販売されているLANalyzer）。

【0022】データ収集ソフトウェア220は、様々なクライアント・システム上の様々なソフトウェア・モジュールの使用の履歴を作成するバックグラウンド・タスクとして実行するように設計される。この履歴は、特定のクライアントが要求したモジュールのモデル、モジュールが要求された時刻、ならびに異なるモジュール・タイプが使用される代表的なシーケンスを得るためにデータ収集タスクによって数日間または数週間にわたってコンパイルされる。

【0023】例えば、クライアントXは、既存の予定された会議の時刻を検査するためにかつ／またはさらに会議情報を追加するために常に営業日の始めにカレンダー・ソフトウェア・モジュールを要求する。この特定のユーザは、次いで前の営業日に関する金融情報または他の情報を更新するためにスプレッドシート・モジュールを常に要求する。別のクライアントYは、最初に電子メール・モジュールを要求し、その後カレンダー・モジュールを要求する。使用履歴は、LANが位置する組織内の個々のユーザの役割に応じてクライアント間で異なる可能性があることを理解されたい。

【0024】この履歴情報は、データ収集ソフトウェアによってコンパイルされ、本発明では、いずれかのクライアント・システムの将来の要件に関して予測を行うために使用される。したがって、クライアントXがカレンダー・モジュールを要求したとき、これはデータ収集タスクによって監視される。制御ソフトウェアは、データ収集ソフトウェアからの情報に基づいて、クライアント・システムXが次に使用する可能性のあるソフトウェア・モジュールのタイプを予測する予測アルゴリズム225を含む。この場合、履歴は、クライアントXが必ずカレンダー・モジュールの後でスプレッドシート・モジュールを要求することを示す。この場合、制御ソフトウェアは、ブッシュ・タスク230にクライアントXのユーザからの要求の前にスプレッドシート・モジュールをクライアントXにブッシュさせる。好ましい構成では、次に予測されるモジュールの転送は、LANトラフィックが事前に定義されたかまたは事前に計算されたしきい値よりも小さいという（LAN解析タスクからのトラフィッ

ク情報に基づく) 決定によって決まる。この構成によれば、ネットワーク上の負荷のバランスを改善することができる。

【0025】代替構成では、様々なクライアントによるモジュールの使用の履歴を作成するデータ収集ソフトウェアはない。代わりに、予測は、ソフトウェア・モジュールのタイプ間の関連のリストに基づく。例えば、クライアントは、最初にテキスト・エディタを要求した場合、次に要求されるモジュールが電子メール通信モジュールであろうと予測することが妥当であろう。電子メール通信モジュールは、次いでクライアントがそのモジュールを要求する前にクライアントにブリエンプティブに転送される。モジュールの関連は、適切な形、例えばテーブルの形でサーバ・システム中に格納される。

【0026】予測モードにかかわらず、予測されたソフトウェア・モジュールは、クライアントによる予測された使用の直前に転送されることが理想的であるが、この場合、モジュールは、例えばサービスを表示するウィンドウがクライアントの画面上にポップアップするようにクライアントによって動的にロードされる。この理想的な構成を実現する能力は、データ収集ソフトウェアの精巧化、任意の時刻におけるネットワーク上の使用可能な帯域幅、ならびにクライアント・ユーザの予測可能性に依存する。動的にロードできるモジュールは、使用後に廃棄される(この場合モジュールはサーバによって再送信されなければならない)か、あるいは後続の呼出しを予測してキャッシュに格納される。

【0027】モジュールは、動的にロードできる必要はなく、むしろユーザによる後続の呼出しのためにクライアントのところのキャッシュ・メモリに転送される。任意の適切な構成を使用して、クライアント中のソフトウェア・モジュールのキャッシングを制御することができる。例えば、使用可能なキャッシュがいっぱいのとき、ソフトウェア・モジュールがFIFO方式でバージされるか、あるいは各モジュールがそれに優先度(例えば、テキスト・エディタがスプレッドシートよりも重要である)を関連付けており、モジュール優先度に従ってキャッシュ中のスペースを解放することができる。

【0028】次に図4を参照すると、本発明の好ましい方法に関連するステップの流れ図が示されている。この方法は、図示のように300のところから始まる。サーバ・システムは、ステップ302で、第1のソフトウェア・モジュール・タイプを接続されたクライアント・システムの1つに送信する。これは、クライアントの要求によるか、あるいはサーバによってブリエンプティブな形で実施される。制御ソフトウェアは、ステップ304で、データ収集ソフトウェアによってコンパイルされた履歴情報に基づいて、多分同じクライアントによって要求されるであろうソフトウェア・モジュールのタイプに関して予測を行う。ステップ306で、制御ソフトウェ

アは、LANトラフィックがあるしきい値よりも小さいかどうかに関する決定を行い、そうであれば、ステップ308で、プッシュ・ソフトウェアに予測したモジュールを問題のクライアントにプッシュさせる。このプロセスはステップ310で終了する。

【0029】既に示したように、特定のクライアントに関連する使用履歴は、そのクライアントのユーザに固有である。予測アルゴリズムの成功は、クライアント・システムが同じユーザによって使用されることに依存する。これは、一般にワークステーション・グループを含むオープン・オフィス環境中でもそうである。そのようなオフィス内の個人には、一般に、その上で作業すべき特定のシステムが割り当てられる。したがって、ユーザAが、例えば毎週月曜日にワークステーションWS10を使用する場合、データ収集ソフトウェアは、ソフトウェア・モジュール使用のパターンが月曜日から月曜日まで比較的一定していることを記す。したがって、サーバ・システムは、月曜日の朝にユーザAが到着する前に予測して適切なソフトウェア・モジュールをワークステーションWS1に供給する。同様に、サーバは、その日の残りの時間中にワークステーションWS1からの要求の前に予測して他のソフトウェア・モジュールを供給する。

【0030】何らかの理由でユーザBが月曜日の朝にワークステーションWS10を使用することを選択し、かつ通常WS10によって要求されないソフトウェア・モジュールに対する要求を行った場合、これはデータ収集ソフトウェアによって記される。これに基づいて、ソフトウェア・モジュールの通常の「月曜日」シーケンスを予測して供給する代わりに、制御ソフトウェアは、予測を行うのを中止するか、あるいは上述の形で純粋にソフトウェア・モジュール・タイプの関連に基づいてソフトウェア使用を予測する。

【0031】本発明はまた、インターネットまたはイントラネットのコンテキストにおいても使用できる。図5は、インターネットまたはイントラネット410を介してそのうちの3つがシステム412、414、416として示されている複数のクライアント・システムに接続されたホスト・システム400を含むそのような構成の概略図である。ホスト・システムは、図1のホストと同じ構成を有する。クライアント・システムは、Java使用可能ブラウザ420、またはホスト・システムから受け取ったJavaアプレット・ソフトウェア・モジュールを実行するJava仮想マシンを含む。各クライアントは、任意選択でソフトウェア・モジュールを一時的に記憶するキャッシュ422を含む。ホスト・システムは、クライアント・システムの個々のIPアドレスを参照して、モジュール使用の履歴をコンパイルし、維持することができる。

【0032】以上、好ましい実施形態について、大容量

記憶装置を有しないクライアント・システムに関して説明してきたが、本発明は、ソフトウェアおよび／またはデータの局所的に記憶するハード・ファイルなど何らかの不揮発性記憶装置または同様の記憶装置を有するクライアント・システムにも適用できることが理解できよう。

【0033】さらに、好ましい実施形態では、単一のサーバ・システムが監視機能および予測機能を実施し、また予測したモジュールをクライアントに転送する。代替構成では、監視ステップおよび予測ステップは1つのサーバのところで引き受けられ、ソフトウェア・モジュールの記憶および転送は第2のサーバのところで引き受けられる。

【0034】まとめとして、本発明の構成に関して以下の事項を開示する。

【0035】(1) データ処理ネットワーク中でソフトウェア・モジュールをターゲット・データ処理システム上で実行するためにホスト・データ処理システムからブリエンブティブに転送する方法であって、それぞれ前記ターゲット・システムのところで異なるサービスを提供する複数のタイプのソフトウェア・モジュールがあり、前記ホスト・システムのところで、ターゲット・システムに転送される第1のソフトウェア・モジュールのタイプを決定するステップと、前記第1のモジュールのタイプの決定に基づいて、次に前記ターゲット・システムのところで必要とされるであろう第2のソフトウェア・モジュールのタイプを予測するステップと、前記第2のタイプのモジュールを前記ターゲット・システムに自動的に転送するステップとを含む方法。

(2) 前記サーバのところで、各ターゲット・システムにおけるソフトウェア・モジュール使用の履歴をコンパイルする他のステップを含み、前記予測ステップが各ターゲット・システムの前記コンパイルされた履歴に基づく上記(1)に記載の方法。

(3) 前記ネットワーク上のトラフィックのレベルを監視し、かつ前記トラフィックがあるしきい値よりも小さいときだけ前記第2のソフトウェア・モジュールを自動的に転送する他のステップを含む上記(1)または(2)に記載の方法。

(4) 前記第1のソフトウェア・モジュールが前記ターゲット・システムからの要求に回答して、前記ホスト・システムから転送される上記(1)ないし(3)のいずれか一項に記載の方法。

(5) 前記ソフトウェア・モジュールが動的にロードできるソフトウェア・モジュールである上記(1)ないし(4)のいずれか一項に記載の方法。

(6) 前記動的にロードできるソフトウェア・モジュールが、前記ターゲット・システム上でJava使用可能ブラウザまたはJava仮想マシンを介して実行するためのJavaアプレットである上記(5)に記載の方

法。

(7) 前記ネットワークがローカル・エリア・ネットワークである上記(1)ないし(6)のいずれか一項に記載の方法。

(8) 前記ネットワークがインターネットまたはイントラネットである上記(1)ないし(6)のいずれか一項に記載の方法。

(9) 前記第1のソフトウェア・モジュールが前記ターゲット・システムのところでテキスト編集サービスを提供する上記(1)ないし(8)のいずれか一項に記載の方法。

(10) 前記第2のソフトウェア・モジュールが前記ターゲット・システムのところで電子メール通信サービスを提供する上記(9)に記載の方法。

(11) 複数のターゲット・データ処理システムと通信するために接続されたサーバ・データ処理システムを含むデータ処理ネットワークであって、前記サーバ・データ処理システムのところにあるか、またはそれに関連し、それぞれターゲット・システムのところで異なるサービスを提供する複数のタイプのソフトウェア・モジュールを前記複数のターゲット・システムにその上で実行するために転送する手段と、前記ターゲット・システムに送信された前記ソフトウェア・モジュールの識別を監視する手段と、第1のタイプのソフトウェア・モジュールがターゲット・システムと通信していたという決定に回答して、第2のタイプのソフトウェア・モジュールが次に前記ターゲット・システムのところで必要とされるであろうと予測し、かつ前記転送手段に前記第2のタイプのソフトウェア・モジュールを前記ターゲット・システムにブリエンブティブに転送させる手段と、前記ターゲット・システムのところで、前記ホスト・システムからの前記ソフトウェア・モジュールを受信する手段とを含むデータ処理ネットワーク。

(12) 各前記ターゲット・システムにおけるソフトウェア・モジュール使用の履歴をコンパイルするデータ収集手段をさらに含み、前記予測手段が前記予測を各ターゲット・システムの前記履歴を基にする上記(11)に記載のデータ処理ネットワーク。

(13) 前記ネットワーク上のトラフィックのレベルを監視し、かつ前記トラフィックがあるしきい値よりも小さいときだけ前記第2のソフトウェア・モジュールを自動的に転送する手段をさらに含む上記(11)または(12)に記載のデータ処理ネットワーク。

(14) 各ソフトウェア・モジュールのタイプが動的にロードできるソフトウェア・モジュールである上記(11)ないし(13)のいずれかに記載のデータ処理ネットワーク。

(15) 前記動的にロードできるソフトウェア・モジュールがJavaアプレットであり、各ターゲット・システムがJava使用可能ブラウザまたはJava仮想マ

13

シンを含む上記(14)に記載のデータ処理ネットワーク。

(16)複数のクライアント・データ処理システムと通信する手段と、それぞれ前記ターゲット・システムのところと異なるサービスを提供する複数のタイプのソフトウェア・モジュールを前記複数のクライアント・システムにその上で実行するために転送する手段と、前記ターゲット・システムに送信された前記ソフトウェア・モジュールの識別を監視する手段と、第1のタイプのソフトウェア・モジュールがターゲット・システムと通信していたという決定にตอบสนองして、第2のタイプのソフトウェア・モジュールが次に前記ターゲット・システムのところと必要とされるであろうと予測し、かつ前記転送手段に前記第2のタイプのソフトウェア・モジュールを前記ターゲット・システムにプリエンブティブに転送させる手段とを含むサーバ・データ処理システム。

【図面の簡単な説明】

【図1】複数のクライアント・データ処理システムに接続されたホスト・システムを含むデータ処理ネットワークの概略図である。

【図2】図1のクライアント・システムおよびホスト・システムの主要な構成要素の概略図である。

【図3】本発明の好ましい実施形態において使用されるホスト・システムの要素間の対話を示す概略図である。

【図4】本発明の好ましい実施形態のステップを示す流れ図である。

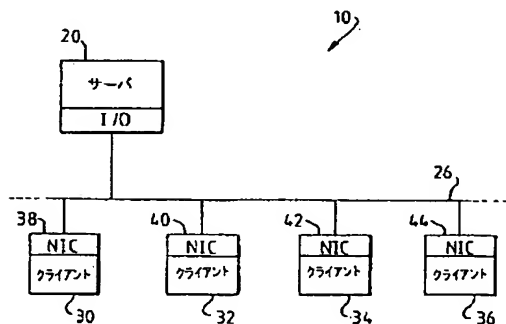
【図5】本発明がそれと共に使用される代替ネットワーク構成の概略図である。

【符号の説明】

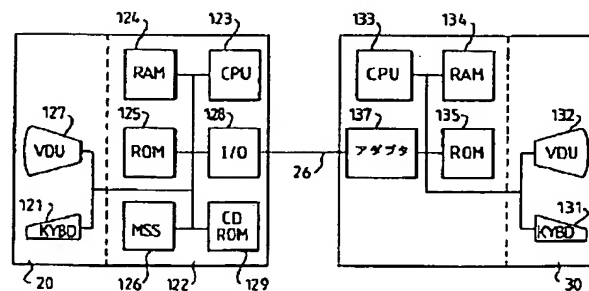
- 10 ローカル・エリア・ネットワーク(LAN)  
20 サーバ・コンピュータ・システム  
26 通信リンク  
30 クライアント・コンピュータ・システム  
32 クライアント・コンピュータ・システム

- \* 34 クライアント・コンピュータ・システム  
36 クライアント・コンピュータ・システム  
38 ネットワーク・インタフェース・カード  
40 ネットワーク・インタフェース・カード  
42 ネットワーク・インタフェース・カード  
44 ネットワーク・インタフェース・カード  
121 キーボード  
122 システム装置  
123 主CPU  
124 システムRAM  
125 システムROM  
126 大容量記憶装置  
127 ディスプレイ  
128 入出力論理回路  
129 CD-ROMドライブ  
131 キーボード  
132 ディスプレイ  
133 主CPU  
134 システム・メモリ(RAM)  
135 不揮発性メモリ(ROM)  
20 ネットワーク・アダプタ・カード  
205 ソフトウェア・コンポーネント  
210 コントローラ  
210 LAN解析ソフトウェア  
220 データ収集ソフトウェア  
225 予測アルゴリズム  
230 ブッシュ・タスク  
400 ホスト・システム  
410 インターネットまたはイントラネット  
412 クライアント・システム  
414 クライアント・システム  
416 クライアント・システム  
420 Java使用可能ブラウザ  
\* 422 キャッシュ

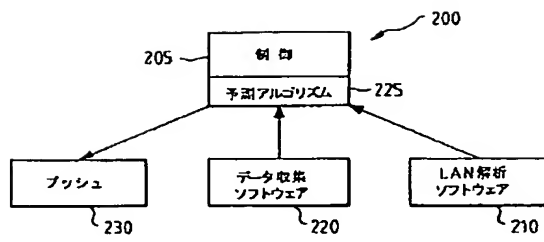
【図1】



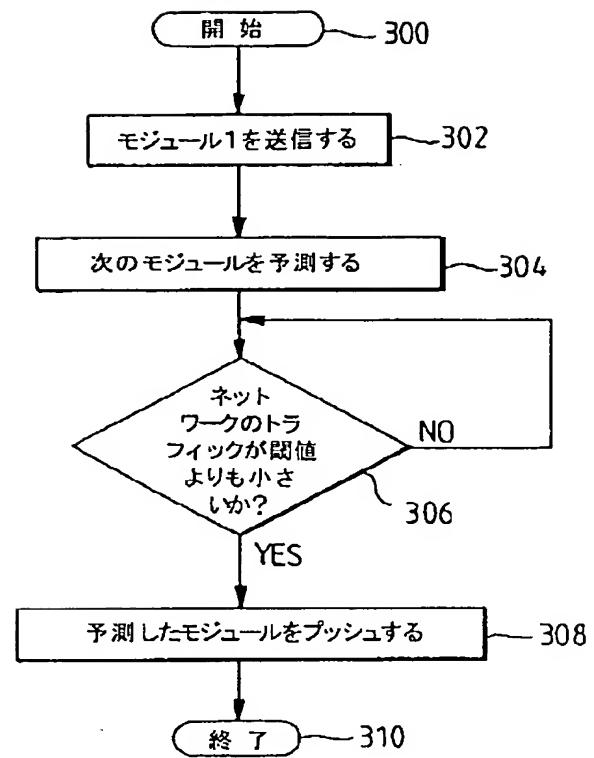
【図2】



【図3】

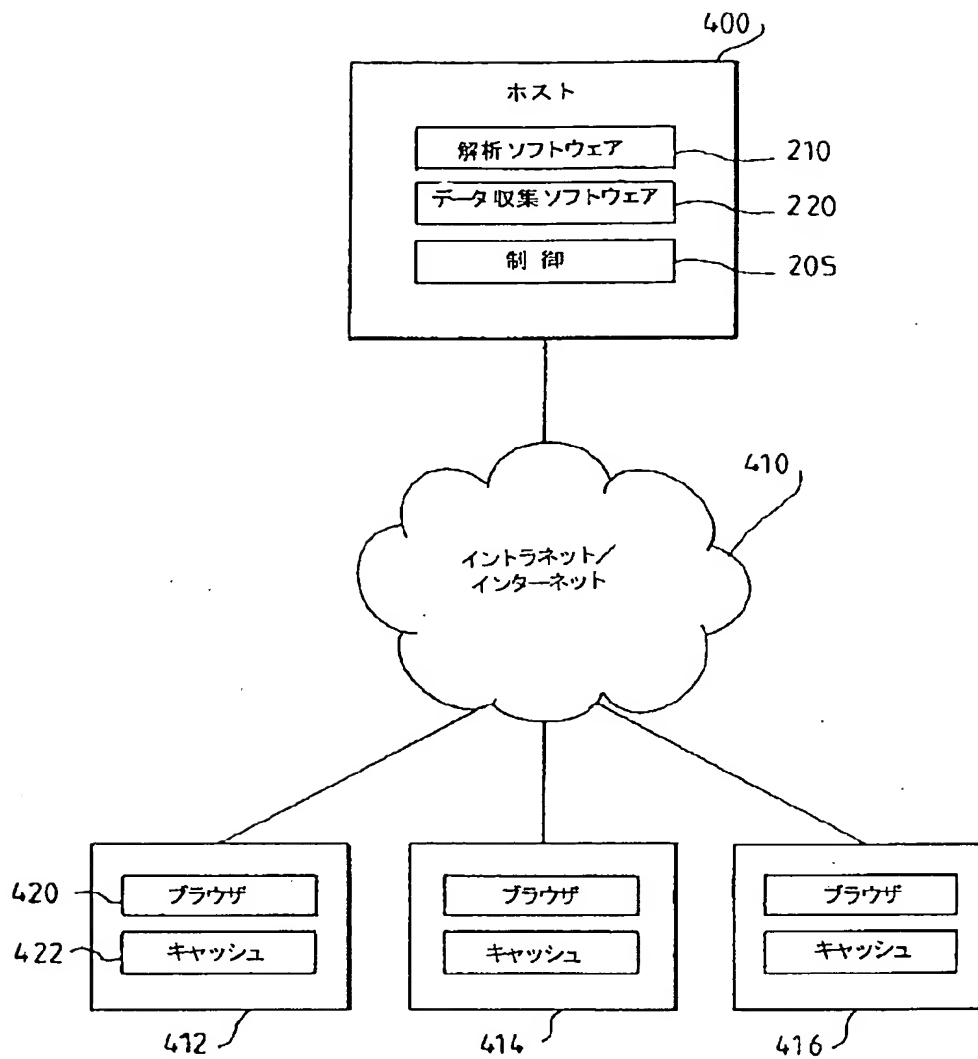


【図4】





【図5】



フロントページの続き

(72)発明者 ソハイル・シード  
イギリス ジー41 5イー・エー ストラ  
ッチクライド グラスゴー ボロックシー  
ルド アルバート・ドライブ 319